

Data format sd_resultat

Summarized:

Contents

1 Introduction	3
2 Typing of a sd_resultat.....	3
3 Structure of the sd_resultat	3.3.1
Parameters and variables of accesses.....	3.3.2
Symbolic name of fields.....	3.3.3
diagrammatic Representation.....	3
4 (under) typing of the sd_resultat.....	5
5 Tree structure.....	5
6 Contents of the basic JEVEUX objects.....	5.6.1
“.DESC”	5.6.2
“.TACH”	5.6.3
“.NOVA”	5.6.4
“.TAVA”	6.6.5
“.ORDR”	6.6.6
“.REFD”	6
7 Rule of construction of the names of the fields.....	6
8 Access rule to the values of the variables of access and the parameters.....	7
9 Symbolic names, variables of access and parameters of the sd_resultat.....	7
10 Example of a sd_resultat of the type MODE_MECA.....	8

1 Introduction

results of one computation by finite elements are fields of scalars, vectors or tensors, but also of the parameters attached to these fields. For example, the modal analysis of a structure results in calculating the eigenvectors (fields of displacement) and the associated eigenfrequencies.

When computation is not reduced to the resolution of only one linear system, the operators produce a set of fields and of parameters which are gathered in "made up" data structure Result described in this document and which is also called to be more precise `sd_resultat`.

2 Typing of a `sd_resultat`

the results (fields and parameters) likely to be stored in a `sd_resultat` are rather variable. For example, the results of a transient dynamic computation can be velocity fields or of acceleration, which is not the case for a quasi-static computation, the results of a thermal computation can be fields of temperatures or heat flux.

To distinguish all the possible situations the `sd_resultat` are typified. One will speak for example about `sd_resultat` of the `dyna_tran` type for the results of a transient dynamic computation, `evol_noli` for a nonlinear quasi-static computation, `evol_ther` for a thermal computation. The `sd_resultat` are create by a single routine `rscrsd` [D6.05.01] whose source is relatively explicit.

All the `sd_resultat` some are their types are accessible to the programming starting from the same routines [D6.05.01].

3 Structure of the `sd_resultat`

3.1 Parameters and variables of access

information (fields and parameters) of a `sd_resultat` is indexed by an integer. This index is called sequence number or `NUME_ORDRE`. This index does not vary inevitably from 1 to N. The sequence numbers can be negative or null; they can not be consecutive. For `RESULTAT` of a transitory type for example sequence number 0 corresponds at the initial state, sequence number 1 corresponds to the first time of computation. A this `NUME_ORDRE` corresponds one or more parameters which also make it possible to reach information. For example, the time of an evolutionary computation or the frequency or the sequence number of an eigen mode. These parameters of a a little particular kind are called variable access. The other parameters (generalized mass of a mode, for example) are simply called parameters without another distinction.

A parameter (variable of access or other) can be of the whole, real, complex type or character.

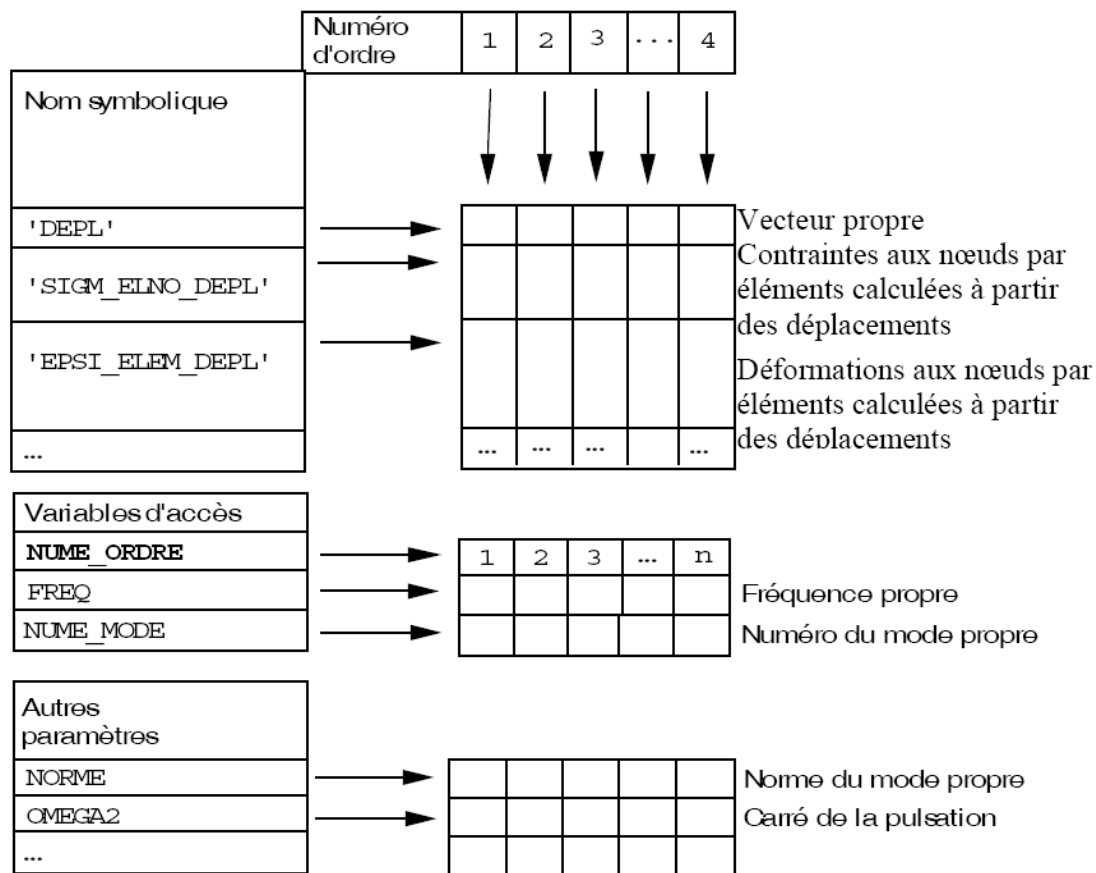
3.2 Symbolic name of the fields

A a `NUME_ORDRE` given can correspond several fields which one distinguishes by what one calls a symbolic name, for example, `DEPL` for the fields of displacements, `SIGM_ELNO` for the stresses by elements with the nodes calculated starting from the field of displacement. These fields are of `cham_no` type, `cham_elem` or `card`.

To simplify the symbolic names retained for the various types of `sd_resultat` are names of quantities or names of computation options.

3.3 Schematic representation

Schematically a `sd_resultat` thus arises in the following form (example of the `sd_resultat` of the `mode_meca` type).



In the first two-dimensional board (symbolic name and sequence number), one finds field names (K19 stored in a vector of K24).

In 2nd and 3rd tables, one finds the value (scalar) variables of access (or parameters).

4 (under) typing of the sd_resultat

the concepts sd_resultat are typified.

The various concepts sd_resultat are as follows (this list nonrestrictive, could be supplemented as developments of Aster; it is not, however not desirable that it lengthens too much):

EVOL_ELAS	Result resulting from a quasi-static computation with evolution with time
EVOL_NOLI	Result resulting from a quasi-static or dynamic computation nonlinear
DYNA_TRANS	Result resulting from a transitory linear dynamic computation in physical space
DYNA_HARMO	Result resulting from a harmonic dynamic computation in physical space
ACOU_HARMO	Result resulting from a harmonic acoustic computation in physical space
MODE_MECA	Result resulting from a computation of searches from eigenvalues and mechanical eigenvectors
MODE_GENE	Result resulting from a computation of searches from eigenvalues and mechanical eigenvectors from generalized quantities
MODE_ACOU	Result resulting from a computation of searches from eigenvalues and mechanical eigenvectors from acoustic quantities
MODE_STAT	Result resulting from a computation of static modes
EVOL_THER	Result resulting from a transitory thermal computation
BASE_MODALE	Result resulting from a regrouping of mechanical modes and static modes

5 Tree structure

```
"sd_resultat" (K19)  :: == record

    (O)      ".DESC" OJB  S  N  K16
             ".TACH" OJB  XC V  K24
             ".NOVA" OJB  S  N  K16
             ".TAVA" OJB  XC V  K8
             ".ORDR" OJB  S  V  I

    # if      MODE_MECA, MODE_GENE, MODE_STAT, BASE_MODALE, ...
    (F)      ".REFD" OJB  S  V  K24  LONG=6
```

6 Contained of basic JEVEUX objects

6.1 ".DESC"

This object is a pointer of name containing symbolic names of these fields.

6.2 ".TACH"

Contains the names of the fields contained in the RESULTAT. This object is a collection of vectors constant length pointed by the .DESC.

6.3 ".NOVA"

This object is a pointer of name containing the names of the variables of access and the parameters of computation.

6.4 “.TAVA”

Describes the variables of access and the parameters of computation. This object is a collection of vectors length equal to 4 pointed by the .NOVA. For a given name, one finds:

1. the name of the suffix of the OBJ where the value (K5) is stored,
2. the characters associated with the row with the parameter making it possible to find its value when a sequence number is associated to him,
3. the nombre total of different parameters contained in l'OBJ,
4. one indicates if it is a variable of access or a parameter.

See example below

6.5 “.ORDR”

This object is a vector of integers.

It contains the sequence numbers stored in the SD.

For example: .ORDR = (0,10,20,30)

This SD has 4 sequence numbers: 0,10,20,30 associates with 4 sequence numbers 1,2,3,4.

6.6 “.REFD”

Vector of 6 K24 (they all are not inevitably used at the same time):

- (1) : name of the MATR_ASSE of stiffness
- (2) : name of the MATR_ASSE of mass
- (3) : name of the MATR_ASSE of damping
- (4) : name of the NUME_DDL
- (5) : name of SD INTERF_DYNA_CLAS

7 Regulates construction of the names of the fields

the name of data structures of the fields contained in the .TACH is composed to leave:

- of the first 8 characters of the name of the “made up” result concept: resu
- of the characters associated with the number in the pointer with symbolic name with result (K3): nusymb
- of the characters (K6) associated at the sequence number: nuordr (limitation with 106 time step) what gives:

```
K8      K1      K3      K1      K6      = K19
resu//". " //nusymb// ". " //nuordr
```

Example:

That is to say “moderesu” a name of “made up” result concept:

- the eigenvector of the mode of sequence number 1 is a cham_no of name “moderesu.001.00001”
- the nodal deformation by element of the mode of sequence number 9 is a cham_elem of name “moderesu.003.00009”

8 Access rule to the values of the variables of access and with the parameters

the value of a variable of access or a parameter of name *nosymb* and sequence number *nuordr* of a result concept "made up" of name *resu* is in the OJB of name:

```
K19 K5 = K24
resu//nosuff
```

with the index $nmax * (irang - 1) + ivar$

where:

irang is the sequence number of the sequence number *nuordr*.

- the name of the suffix *nosuff*, the *nmax* number of variables, and the index *ivar* are respectively in the first, second and third elements of the vector of the *nosymb* object of the collection of name:

```
K19 K5 = K24
resu//".TAVA"
```

Example:

That is to say "moderesu" is a name of "composed" result concept, generalized stiffness *RIGI_GENE* of the mode of sequence number 11 (arranged with index 7 of object *.ORDER*) will be in the OJB vector:

"moderesu .PARAM"

with the index $27 * (7 - 1) + 5$

".PARAM", "27" and "5" were found in the OJB of name "moderesu .TAVA" opposite name *RIGI_GENE*.

9 Symbolic names, variables of access and parameters of the *sd_resultat*

One gives here as an indication quelques parameter and field names stored in an *EVOL_ELAS*.

```
* Symbolic names of the fields:
Variable "DEPL" "SIEF_ELGA" "SIEF_ELGA_DPGE"
"EPSI_ELNO" "EPSI_ELNO_DPGE" "DEGE_ELNO"
"
```

```
... * of access:
"INST"
```

```
* Parameters:
"EFFORT_N" "MOMENT_MFY" "MOMENT_MFZ"
```

...

10 Example of a sd_resultat of the type MODE_MECA

